ESTIMATION OF HIGH-DIMENSIONAL DIFFERENTIAL GRAPHS FROM MULTI-ATTRIBUTE DATA

Jitendra K. Tugnait

Department of Electrical & Computer Engineering
Auburn University, Auburn, AL 36849, USA

ABSTRACT

We consider the problem of estimating differences in two Gaussian graphical models (GGMs) which are known to have similar structure. The GGM structure is encoded in its precision (inverse covariance) matrix. In many applications one is interested in estimating the difference in two precision matrices to characterize underlying changes in conditional dependencies of two sets of data. Existing methods for differential graph estimation are based on single-attribute models where one associates a scalar random variable with each node. In multi-attribute graphical models, each node represents a random vector. In this paper, we analyze a group lasso penalized D-trace loss function approach for differential graph learning from multi-attribute data. An alternating direction method of multipliers (ADMM) algorithm is presented to optimize the objective function. Theoretical analysis establishing consistency in support recovery and estimation in high-dimensional settings is provided. We illustrate our approach using a numerical example where the multi-attribute approach is shown to outperform a single-attribute approach.

Keywords: Sparse graph learning; differential graph estimation; undirected graph; multi-attribute graphs.

1. INTRODUCTION

Graphical models provide a powerful tool for analyzing multivariate data [1, 2]. In a statistical graphical model, the conditional statistical dependency structure among p random variables $x_1, x_1, \cdots, x_p$, is represented using an undirected graph $\mathcal{G} = (V, \mathcal{E})$. The graph $\mathcal{G}$ then is a conditional independence graph (CIG) where there is no edge between nodes i and j (i.e., $\{i,j\} \not\in \mathcal{E}$) iff x_i and x_j are conditionally independent given the remaining p-2 variables x_ℓ, $\ell \in [p]$, $\ell \neq i$, $\ell \neq j$. In particular, Gaussian graphical models (GGMs) are CIGs where $\boldsymbol{x}$ is multivariate Gaussian. Suppose $\boldsymbol{x}$ has positive-definite covariance matrix $\boldsymbol{\Sigma}$ with inverse covariance matrix $\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1}$. Then Ω_{ij}, the (i,j)-th element of $\boldsymbol{\Omega}$, is zero iff x_i and x_j are conditionally independent. Such models for $\boldsymbol{x}$ have been extensively studied. Given n samples of $\boldsymbol{x}$, in *high-dimensional settings* where $p \gg 1$ and/or n is of the order of p, one estimates $\boldsymbol{\Omega}$ under some sparsity constraints; see [3–6]. More recently there has been increasing interest in differential network analysis where one is interested in estimating the difference in two inverse covariance matrices [9–11]. Given observations $\boldsymbol{x}$ and $\boldsymbol{y}$ from two groups of subjects, one is interested in the difference $\boldsymbol{\Delta} = \boldsymbol{\Omega}_y - \boldsymbol{\Omega}_x$, where $\boldsymbol{\Omega}_x = (E\{\boldsymbol{x}\boldsymbol{x}^\top\})^{-1}$ and $\boldsymbol{\Omega}_y = (E\{\boldsymbol{y}\boldsymbol{y}^\top\})^{-1}$. The associated differential graph is $\mathcal{G}_\Delta = (V, \mathcal{E}_\Delta)$ where $\{i,j\} \in \mathcal{E}_\Delta$ iff $\boldsymbol{\Delta}_{ij} \neq 0$. It characterizes differences between the GGMs of the two sets of data. We use the term differential graph as in [7, 8] ([9–11] use the term differential network). As noted in [11], in biostatistics, the differential network/graph describes the changes in conditional dependencies between components under different environmental or genetic conditions. For instance, one may be interested in the differences in the graphical models of healthy and impaired subjects, or models under different disease states, given gene expression data or functional MRI signals [3, 12, 13].

In the preceding graphs, each node represents a scalar random variable. In many applications, there may be more than one random variable associated with a node. This class of graphical models has been called multi-attribute graphical models in [14–17] and vector graphs or networks in [18–21]. In a gene regulatory network, one may have different molecular profiles available for a single gene, such as protein, DNA and RNA. Since these molecular profiles are on the same set of biological samples, they constitute multi-attribute data for gene regulatory graphical models in [14, 16]. Consider p jointly Gaussian vectors $\boldsymbol{z}_i \in \mathbb{R}^m$, $i \in [p]$. We associate $\boldsymbol{z}_i$ with the ith node of graph $\mathcal{G} = (V, \mathcal{E})$, $V = [p]$, $\mathcal{E} \subseteq V \times V$. We now have m attributes per node. Now $\{i,j\} \in \mathcal{E}$ iff vectors $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$ are conditionally independent given the remaining p-2 vectors $\{\boldsymbol{z}_\ell, \ell \in V \backslash \{i,j\}\}$. Let $\boldsymbol{x} = [\boldsymbol{z}_1^\top \; \boldsymbol{z}_2^\top \; \cdots \; \boldsymbol{z}_p^\top]^\top \in \mathbb{R}^{mp}$. Let $\boldsymbol{\Omega} = (E\{\boldsymbol{x}\boldsymbol{x}^\top\})^{-1}$ assuming $E\{\boldsymbol{x}\boldsymbol{x}^\top\} \succ \mathbf{0}$. Define the $m \times m$ subblock $\boldsymbol{\Omega}^{(ij)}$ of $\boldsymbol{\Omega}$ as $[\boldsymbol{\Omega}^{(ij)}]_{rs} = [\boldsymbol{\Omega}]_{(i-1)m+r,(j-1)m+s}$, $r, s = 1, 2, \cdots, m$. Then we have the following equivalence [16, Sec. 2.1]

$$\{i,j\} \not\in \mathcal{E} \;\Leftrightarrow\; \boldsymbol{\Omega}^{(ij)} = \mathbf{0}\,. \tag{1}$$

This paper is concerned with estimation of differential graphs from multi-attribute data. Given samples $\boldsymbol{x}(t)$, $t = 1, 2, \cdots, n_x$, of $\boldsymbol{x} = [\boldsymbol{z}_1^\top \; \boldsymbol{z}_2^\top \; \cdots \; \boldsymbol{z}_p^\top]^\top \in \mathbb{R}^{mp}$ where $\boldsymbol{z}_i \in \mathbb{R}^m$, $i \in [p]$, are jointly Gaussian, and similarly given samples $\boldsymbol{y}(t)$, $t = 1, 2, \cdots, n_y$, of $\boldsymbol{y} \in \mathbb{R}^{mp}$, our objective is to estimate the difference $\boldsymbol{\Delta} = \boldsymbol{\Omega}_y - \boldsymbol{\Omega}_x$, and determine the differential graph $\mathcal{G}_\Delta = (V, \mathcal{E}_\Delta)$ with edgeset $\mathcal{E}_\Delta = \{\{k,\ell\} \, : \, \|\boldsymbol{\Delta}^{(k\ell)}\|_F \neq 0\}$. Multi-attribute differential graphs have not been investigated before. The work of [7, 8] is similar to a multi-attribute formulation except that in [7, 8] $\boldsymbol{x}(t)$ and $\boldsymbol{y}(t)$ are non-stationary ("functional" modeling), and instead of a single record (sample) of $\boldsymbol{x}(t)$, $t = 1, 2, \cdots, n_x$ and $\boldsymbol{y}(t)$, $t = 1, 2, \cdots, n_y$, as in this paper, they assume multiple independent observations of $\boldsymbol{x}(t)$, $t \in \mathcal{T}$, and $\boldsymbol{y}(t)$, $t \in \mathcal{T}$.

Notation: For a set V, $|V|$ or $\text{card}(V)$ denotes its cardinality. Given $\boldsymbol{A} \in \mathbb{R}^{p \times p}$, we use $\phi_{\min}(\boldsymbol{A})$, $\phi_{\max}(\boldsymbol{A})$, $|\boldsymbol{A}|$ and $\text{tr}(\boldsymbol{A})$ to denote the minimum eigenvalue, maximum eigenvalue, determinant and trace of $\boldsymbol{A}$, respectively. For $\boldsymbol{B} \in \mathbb{R}^{p \times q}$, we define $\|\boldsymbol{B}\| = \sqrt{\phi_{\max}(\boldsymbol{B}^\top \boldsymbol{B})}$, $\|\boldsymbol{B}\|_F = \sqrt{\text{tr}(\boldsymbol{B}^\top \boldsymbol{B})}$, $\|\boldsymbol{B}\|_1 = \sum_{i,j} |B_{ij}|$, where B_{ij} is the (i,j)-th element of $\boldsymbol{B}$ (also denoted by $[\boldsymbol{B}]_{ij}$), $\|\boldsymbol{B}\|_\infty = \max_{i,j} |B_{ij}|$ and $\|\boldsymbol{B}\|_{1,\infty} = \max_i \sum_j |B_{ij}|$. The symbols $\otimes$ and $\boxtimes$ denote Kronecker product and Tracy-Singh product [22], respectively. In particular, given block partitioned

This work was supported by NSF Grant ECCS-2040536. Author's email: tugnajk@auburn.edu

matrices $\boldsymbol{A} = [\boldsymbol{A}_{ij}]$ and $\boldsymbol{B} = [\boldsymbol{B}_{k\ell}]$ with submatrices $\boldsymbol{A}_{ij}$ and $\boldsymbol{B}_{k\ell}$, Tracy-Singh product yields another block partitioned matrix $\boldsymbol{A} \boxtimes \boldsymbol{B} = [\boldsymbol{A}_{ij} \boxtimes \boldsymbol{B}]_{ij} = [[\boldsymbol{A}_{ij} \otimes \boldsymbol{B}_{k\ell}]_{k\ell}]_{ij}$ [23]. Given $\boldsymbol{A} = [\boldsymbol{A}_{ij}] \in \mathbb{R}^{mp \times mp}$ with $\boldsymbol{A}_{ij} \in \mathbb{R}^{m \times m}$, $\text{vec}(\boldsymbol{A}) \in \mathbb{R}^{m^2p^2}$ denotes the vectorization of $\boldsymbol{A}$ which stacks the columns of the matrix $\boldsymbol{A}$, and $\text{bvec}(\boldsymbol{A}) \in \mathbb{R}^{m^2p^2}$ is given by $\text{bvec}(\boldsymbol{A}) = [(\text{vec}(\boldsymbol{A}_{11}))^\top \ (\text{vec}(\boldsymbol{A}_{21}))^\top \ \cdots \ (\text{vec}(\boldsymbol{A}_{p1}))^\top \ (\text{vec}(\boldsymbol{A}_{12}))^\top \ \cdots \ (\text{vec}(\boldsymbol{A}_{p2}))^\top \ \cdots \ (\text{vec}(\boldsymbol{A}_{pp}))^\top]^\top$. Let $S = \mathcal{E}_\Delta = \{\{k, \ell\} : \|\boldsymbol{\Delta}^{(k\ell)}\|_F \neq 0\}$ where $\boldsymbol{\Delta} = [\boldsymbol{\Delta}^{(k\ell)}] \in \mathbb{R}^{mp \times mp}$ with $\boldsymbol{\Delta}^{(k\ell)} \in \mathbb{R}^{m \times m}$. Then $\boldsymbol{\Delta}_S$ denotes the submatrix of $\boldsymbol{\Delta}$ with block rows and columns indexed by S, i.e., $\boldsymbol{\Delta}_S = [\boldsymbol{\Delta}^{(k\ell)}]_{(k,\ell) \in S}$. Suppose $\boldsymbol{\Gamma} = \boldsymbol{A} \boxtimes \boldsymbol{B}$ given block partitioned matrices $\boldsymbol{A} = [\boldsymbol{A}_{ij}]$ and $\boldsymbol{B} = [\boldsymbol{B}_{k\ell}]$. For any two subsets T_1 and T_2 of $[p] \times [p]$, $\boldsymbol{\Gamma}_{T_1,T_2}$ denotes the submatrix of $\boldsymbol{\Gamma}$ with block rows and columns indexed by T_1 and T_2, i.e., $\boldsymbol{\Gamma}_{T_1,T_2} = [\boldsymbol{A}_{j\ell} \otimes \boldsymbol{B}_{kq}]_{(j,k) \in T_1, (\ell,q) \in T_2}$.

2. GROUP LASSO PENALIZED D-TRACE LOSS

Let $\boldsymbol{x} = [\boldsymbol{z}_1^\top \ \boldsymbol{z}_2^\top \ \cdots \ \boldsymbol{z}_p^\top]^\top \in \mathbb{R}^{mp}$ where $\boldsymbol{z}_i \in \mathbb{R}^m$, $i \in [p]$, are zero-mean, jointly Gaussian. Given i.i.d. samples $\boldsymbol{x}(t)$, $t = 1, 2, \cdots, n_x$, of $\boldsymbol{x}$, and similarly given i.i.d. samples $\boldsymbol{y}(t)$, $t = 1, 2, \cdots, n_y$, of $\boldsymbol{y} \in \mathbb{R}^{mp}$, form the sample covariance estimates

$$\hat{\boldsymbol{\Sigma}}_x = \frac{1}{n_x} \sum_{t=1}^{n_x} \boldsymbol{x}(t)\boldsymbol{x}^\top(t) \,, \quad \hat{\boldsymbol{\Sigma}}_y = \frac{1}{n_y} \sum_{t=1}^{n_y} \boldsymbol{y}(t)\boldsymbol{y}^\top(t) \,. \tag{2}$$

and denote their true values as $\boldsymbol{\Sigma}_x^* = \boldsymbol{\Omega}_x^{-*} (= (\boldsymbol{\Omega}_x^*)^{-1})$ and $\boldsymbol{\Sigma}_y^* = \boldsymbol{\Omega}_y^{-*}$. We wish to estimate $\boldsymbol{\Delta} = \boldsymbol{\Omega}_y^* - \boldsymbol{\Omega}_x^*$ and graph $\mathcal{G}_\Delta = (V, \mathcal{E}_\Delta)$, based on $\hat{\boldsymbol{\Sigma}}_x$ and $\hat{\boldsymbol{\Sigma}}_y$. Following the single-attribute formulation of [10] (see also [24, Sec. 2.1]), we will use a convex D-trace loss function given by

$$L(\boldsymbol{\Delta}, \hat{\boldsymbol{\Sigma}}_x, \hat{\boldsymbol{\Sigma}}_y) = \frac{1}{2}\text{tr}(\hat{\boldsymbol{\Sigma}}_x \boldsymbol{\Delta} \hat{\boldsymbol{\Sigma}}_y \boldsymbol{\Delta}^\top) - \text{tr}(\boldsymbol{\Delta}(\hat{\boldsymbol{\Sigma}}_x - \hat{\boldsymbol{\Sigma}}_y)) \tag{3}$$

where D-trace refers to difference-in-trace loss function, a term coined in [25] in the context of graphical model estimation. The function $L(\boldsymbol{\Delta}, \boldsymbol{\Sigma}_x^*, \boldsymbol{\Sigma}_y^*)$ is strictly convex in $\boldsymbol{\Delta}$ and has a unique minimum at $\boldsymbol{\Delta}^* = \boldsymbol{\Omega}_y^* - \boldsymbol{\Omega}_x^*$ [10, 24]. When we use sample covariances, we propose to estimate $\boldsymbol{\Delta}$ by minimizing the group-lasso penalized loss function

$$L_\lambda(\boldsymbol{\Delta}, \hat{\boldsymbol{\Sigma}}_x, \hat{\boldsymbol{\Sigma}}_y) = L(\boldsymbol{\Delta}, \hat{\boldsymbol{\Sigma}}_x, \hat{\boldsymbol{\Sigma}}_y) + \lambda \sum_{k,\ell=1}^{p} \|\boldsymbol{\Delta}^{(k\ell)}\|_F \tag{4}$$

where $\lambda > 0$ is a tuning parameter and $\|\boldsymbol{\Delta}^{(k\ell)}\|_F$ promotes blockwise sparsity in $\boldsymbol{\Delta}$ [26–28] where, if we partition $\boldsymbol{\Delta}$ into $m \times m$ submatrices, $\boldsymbol{\Delta}^{(k\ell)}$ denotes its (k, ℓ)th submatrix, associated with edge $\{k, \ell\}$ of the differential graph $\mathcal{G}_\Delta = (V, \mathcal{E}_\Delta)$. Lasso penalty has been used in [10, 24] for single-attribute models.

Suppose

$$\hat{\boldsymbol{\Delta}} = \arg\min_{\boldsymbol{\Delta}} L_\lambda(\boldsymbol{\Delta}, \hat{\boldsymbol{\Sigma}}_x, \hat{\boldsymbol{\Sigma}}_y) \,. \tag{5}$$

Even though $\boldsymbol{\Delta}$ is symmetric, $\hat{\boldsymbol{\Delta}}$ is not. We can symmetrize it by setting $\hat{\boldsymbol{\Delta}}_{sym} = \frac{1}{2}(\hat{\boldsymbol{\Delta}} + \hat{\boldsymbol{\Delta}}^\top)$, after obtaining $\hat{\boldsymbol{\Delta}}$.

3. OPTIMIZATION

Similar to [24] (also [10]), we use an alternating direction method of multipliers (ADMM) approach [29] with variable splitting. Using variable splitting, consider

$$\min_{\boldsymbol{\Delta}, \boldsymbol{W}} \Big\{ L(\boldsymbol{\Delta}, \hat{\boldsymbol{\Sigma}}_x, \hat{\boldsymbol{\Sigma}}_y) + \lambda \sum_{k,\ell=1}^{p} \|\boldsymbol{W}^{(k\ell)}\|_F \Big\} \text{ subject to } \boldsymbol{\Delta} = \boldsymbol{W} \,. \tag{6}$$

The **scaled** augmented Lagrangian for this problem is [29]

$$L_\rho = L(\boldsymbol{\Delta}, \hat{\boldsymbol{\Sigma}}_x, \hat{\boldsymbol{\Sigma}}_y) + \lambda \sum_{k,\ell=1}^{p} \|\boldsymbol{W}^{(k\ell)}\|_F + \frac{\rho}{2}\|\boldsymbol{\Delta} - \boldsymbol{W} + \boldsymbol{U}\|_F^2 \tag{7}$$

where $\boldsymbol{U}$ is the dual variable, and $\rho > 0$ is the penalty parameter. Given the results $\boldsymbol{\Delta}^{(i)}, \boldsymbol{W}^{(i)}, \boldsymbol{U}^{(i)}$ of the ith iteration, in the $(i+1)$st iteration, an ADMM algorithm executes the following three updates:

(a) $\boldsymbol{\Delta}^{(i+1)} \leftarrow \arg\min_{\boldsymbol{\Delta}} L_a(\boldsymbol{\Delta}), \ \ L_a(\boldsymbol{\Delta}) := L(\boldsymbol{\Delta}, \hat{\boldsymbol{\Sigma}}_x, \hat{\boldsymbol{\Sigma}}_y) + \frac{\rho}{2}\|\boldsymbol{\Delta} - \boldsymbol{W}^{(i)} + \boldsymbol{U}^{(i)}\|_F^2$

(b) $\boldsymbol{W}^{(i+1)} \leftarrow \arg\min_{\boldsymbol{W}} L_b(\boldsymbol{W}), \ \ L_b(\boldsymbol{W}) := \lambda \sum_{k,\ell=1}^{p} \|\boldsymbol{W}^{(k\ell)}\|_F + \frac{\rho}{2}\|\boldsymbol{\Delta}^{(i+1)} - \boldsymbol{W} + \boldsymbol{U}^{(i)}\|_F^2$

(c) $\boldsymbol{U}^{(i+1)} \leftarrow \boldsymbol{U}^{(i)} + \Big(\boldsymbol{\Delta}^{(i+1)} - \boldsymbol{W}^{(i+1)}\Big)$

Update (a): Differentiate $L_a(\boldsymbol{\Delta})$ w.r.t. $\boldsymbol{\Delta}$ to obtain

$$\begin{aligned} \boldsymbol{0} = & \frac{\partial L_a(\boldsymbol{\Delta})}{\partial \boldsymbol{\Delta}} = \hat{\boldsymbol{\Sigma}}_x \boldsymbol{\Delta} \hat{\boldsymbol{\Sigma}}_y - (\hat{\boldsymbol{\Sigma}}_x - \hat{\boldsymbol{\Sigma}}_y) + \rho(\boldsymbol{\Delta} - \boldsymbol{W} + \boldsymbol{U}) \quad (8) \\ \Rightarrow & (\hat{\boldsymbol{\Sigma}}_y \otimes \hat{\boldsymbol{\Sigma}}_x + \rho \boldsymbol{I})\text{vec}(\boldsymbol{\Delta}) = \text{vec}(\hat{\boldsymbol{\Sigma}}_x - \hat{\boldsymbol{\Sigma}}_y + \rho(\boldsymbol{W} - \boldsymbol{U})) \quad (9) \end{aligned}$$

Direct matrix inversion solution of (9) requires inversion of a $(mp)^2 \times (mp)^2$ matrix. A computationally cheaper solution is given in [10, 24], as follows. Carry out eigendecomposition of $\hat{\boldsymbol{\Sigma}}_x$ and $\hat{\boldsymbol{\Sigma}}_y$ as $\hat{\boldsymbol{\Sigma}}_x = \boldsymbol{Q}_x \boldsymbol{D}_x \boldsymbol{Q}_x^\top$, $\boldsymbol{Q}_x \boldsymbol{Q}_x^\top = \boldsymbol{I}$ and $\hat{\boldsymbol{\Sigma}}_y = \boldsymbol{Q}_y \boldsymbol{D}_y \boldsymbol{Q}_y^\top$, $\boldsymbol{Q}_y \boldsymbol{Q}_y^\top = \boldsymbol{I}$, where $\boldsymbol{D}_x$ and $\boldsymbol{D}_y$ are diagonal matrices. Then $\hat{\boldsymbol{\Delta}}$ that minimizes $L_a(\boldsymbol{\Delta})$ is given by

$$\hat{\boldsymbol{\Delta}} = \boldsymbol{Q}_x \Big[\boldsymbol{B} \circ [\boldsymbol{Q}_x^\top (\hat{\boldsymbol{\Sigma}}_x - \hat{\boldsymbol{\Sigma}}_y + \rho(\boldsymbol{W} - \boldsymbol{U}))\boldsymbol{Q}_y] \Big] \boldsymbol{Q}_y^\top \tag{10}$$

where the symbol $\circ$ denotes the Hadamard product and $\boldsymbol{B} \in \mathbb{R}^{mp \times mp}$ organizes the diagonal of $(\boldsymbol{D}_y \otimes \boldsymbol{D}_x + \rho \boldsymbol{I})^{-1}$ in a matrix with $\boldsymbol{B}_{jk} = 1/([\boldsymbol{D}_x]_{jj}[\boldsymbol{D}_y]_{kk} + \rho)$. Note that the eigendecomposition of $\hat{\boldsymbol{\Sigma}}_x$ and $\hat{\boldsymbol{\Sigma}}_y$ has to be done only once. Thus

$$\boldsymbol{\Delta}^{(i+1)} = \boldsymbol{Q}_x \Big[\boldsymbol{B} \circ [\boldsymbol{Q}_x^\top (\hat{\boldsymbol{\Sigma}}_x - \hat{\boldsymbol{\Sigma}}_y + \rho(\boldsymbol{W}^{(i)} - \boldsymbol{U}^{(i)}))\boldsymbol{Q}_y] \Big] \boldsymbol{Q}_y^\top \tag{11}$$

Update (b): Here we have the group lasso solution [26–28]

$$\begin{aligned} & (\boldsymbol{W}^{(k\ell)})^{(i+1)} \\ = & \Big(1 - \frac{(\lambda/\rho)}{\|(\boldsymbol{\Delta}^{(i+1)} + \boldsymbol{U}^{(i)})^{(k\ell)}\|_F}\Big)_+ (\boldsymbol{\Delta}^{(i+1)} + \boldsymbol{U}^{(i)})^{(k\ell)} \end{aligned} \tag{12}$$

where $(a)_+ = \max(0, a)$.

Convergence. A stopping (convergence) criterion following [29, Sec. 3.3.1] can be devised. The stopping criterion is based on primal and dual residuals being small where, in our case, at $(i+1)$st iteration, the primal residual is given by $\boldsymbol{\Delta}^{(i+1)} - \boldsymbol{W}^{(i+1)}$ and the dual residual by $\rho(\boldsymbol{W}^{(i+1)} - \boldsymbol{W}^{(i)})$. Convergence criterion is met when the norms of these residuals are below some threshold. The objective function $L_\lambda(\boldsymbol{\Delta}, \hat{\boldsymbol{\Sigma}}_x, \hat{\boldsymbol{\Sigma}}_y)$, given by (4), is strictly convex. It is also closed, proper and lower semi-continuous. Hence, for any fixed $\rho > 0$, the ADMM algorithm is guaranteed to converge [29, Sec. 3.2], in the sense that we have primal residual convergence to 0, dual residual convergence to 0, and objective function convergence to the optimal value.

Model Selection. Following the lasso penalty work of [10] (who invokes [12]), we will use the following criterion for selection of group lasso penalty:

$$\begin{aligned} BIC(\lambda) =&(n_x+n_y)\,\|\hat{\boldsymbol\Sigma}_x\hat{\boldsymbol\Delta}\hat{\boldsymbol\Sigma}_y - (\hat{\boldsymbol\Sigma}_x - \hat{\boldsymbol\Sigma}_y)\|_F \\ &+ \ln(n_x+n_y)\,|\hat{\boldsymbol\Delta}|_0 \end{aligned} \tag{13}$$

where $|\boldsymbol{A}|_0$ denotes number of nonzero elements in $\boldsymbol{A}$ and $\hat{\boldsymbol\Delta}$ obeys (5). Choose λ to minimize $BIC(\lambda)$. Following [10] we term it BIC (Bayesian information criterion) even though the cost function used is not negative log-likelihood although $\ln(n_x+n_y)\,|\hat{\boldsymbol\Delta}|_0$ penalizes over-parametrization as in BIC. It is based on the fact that true $\boldsymbol\Delta^*$ satisfies $\boldsymbol\Sigma_x^*\boldsymbol\Delta^*\boldsymbol\Sigma_y^* - (\boldsymbol\Sigma_x^* - \boldsymbol\Sigma_y^*) = \mathbf{0}$

In our simulations we search over $\lambda \in [\lambda_\ell, \lambda_u]$, where λ_ℓ and λ_u are selected via a heuristic as in [17]. Find the smallest λ, labeled λ_{sm} for which we get a no-edge model; then we set $\lambda_u = \lambda_{sm}/2$ and $\lambda_\ell = \lambda_u/10$.

4. THEORETICAL ANALYSIS

Here we analyze the properties of $\hat{\boldsymbol\Delta}$ by following the approach(es) of [10, 16, 24, 25, 30]. Define the true differential edgeset

$$S = \mathcal{E}_{\Delta^*} = \{\{k,\ell\} \,:\, \|\boldsymbol\Delta^{*(k\ell)}\|_F \neq 0\}\,, \quad s = |S|\,. \tag{14}$$

A necessary and sufficient condition for minimization of convex $L_\lambda(\boldsymbol\Delta, \hat{\boldsymbol\Sigma}_x, \hat{\boldsymbol\Sigma}_y)$ given by (4) w.r.t. $\boldsymbol\Delta \in \mathbb{R}^{mp\times mp}$ is that $\hat{\boldsymbol\Delta}$ minimizes (4) iff the zero matrix belongs to the sub-differential of $L_\lambda(\boldsymbol\Delta, \hat{\boldsymbol\Sigma}_x, \hat{\boldsymbol\Sigma}_y)$. That is,

$$\begin{aligned} \mathbf{0} =& \frac{\partial L_\lambda(\boldsymbol\Delta, \hat{\boldsymbol\Sigma}_x, \hat{\boldsymbol\Sigma}_y)}{\partial \boldsymbol\Delta} + \lambda \boldsymbol{Z}(\boldsymbol\Delta)\Big|_{\boldsymbol\Delta=\hat{\boldsymbol\Delta}} \\ =& \hat{\boldsymbol\Sigma}_x\hat{\boldsymbol\Delta}\hat{\boldsymbol\Sigma}_y - (\hat{\boldsymbol\Sigma}_x - \hat{\boldsymbol\Sigma}_y) + \lambda\boldsymbol{Z}(\hat{\boldsymbol\Delta}) \end{aligned} \tag{15}$$

where $\boldsymbol{Z}(\boldsymbol\Delta) \in \partial \sum_{k,\ell=1}^p \|\boldsymbol\Delta^{(k\ell)}\|_F \in \mathbb{R}^{mp\times mp}$, the sub-differential of group lasso penalty term, is given by

$$(\boldsymbol{Z}(\boldsymbol\Delta))^{(k\ell)} = \begin{cases} \frac{\boldsymbol\Delta^{(k\ell)}}{\|\boldsymbol\Delta^{(k\ell)}\|_F} & \text{if } \|\boldsymbol\Delta^{(k\ell)}\|_F \neq 0 \\ \boldsymbol{V} \in \mathbb{R}^{m\times m},\ \|\boldsymbol{V}\|_F \le 1, & \text{if } \|\boldsymbol\Delta^{(k\ell)}\|_F = 0 \end{cases} \tag{16}$$

In terms of $m \times m$ submatrices of $\boldsymbol\Delta$, $\hat{\boldsymbol\Sigma}_x$, $\hat{\boldsymbol\Sigma}_y$ and $\boldsymbol{Z}(\boldsymbol\Delta)$ corresponding to various graph edges, using $\text{bvec}(\boldsymbol{ADB}) = (\boldsymbol{B}^\top \boxtimes \boldsymbol{A})\text{bvec}(\boldsymbol{D})$ [22, Lemma 1], we may rewrite (15) as

$$(\hat{\boldsymbol\Sigma}_y \boxtimes \hat{\boldsymbol\Sigma}_x)\text{bvec}(\hat{\boldsymbol\Delta}) - \text{bvec}(\hat{\boldsymbol\Sigma}_x - \hat{\boldsymbol\Sigma}_y) + \lambda\,\text{bvec}(\boldsymbol{Z}(\hat{\boldsymbol\Delta})) = \mathbf{0} \tag{17}$$

Define

$$\boldsymbol\Gamma^* = \boldsymbol\Sigma_y^* \boxtimes \boldsymbol\Sigma_x^*\,, \quad \hat{\boldsymbol\Gamma} = \hat{\boldsymbol\Sigma}_y \boxtimes \hat{\boldsymbol\Sigma}_x\,. \tag{18}$$

Then (17) can be rewritten as

$$\begin{aligned} &\begin{bmatrix} \hat{\boldsymbol\Gamma}_{S,S} & \hat{\boldsymbol\Gamma}_{S,S^c} \\ \hat{\boldsymbol\Gamma}_{S^c,S} & \hat{\boldsymbol\Gamma}_{S^c,S^c} \end{bmatrix} \begin{bmatrix} \text{bvec}(\hat{\boldsymbol\Delta}_S) \\ \text{bvec}(\hat{\boldsymbol\Delta}_{S^c}) \end{bmatrix} - \begin{bmatrix} \text{bvec}((\hat{\boldsymbol\Sigma}_x - \hat{\boldsymbol\Sigma}_y)_S) \\ \text{bvec}((\hat{\boldsymbol\Sigma}_x - \hat{\boldsymbol\Sigma}_y)_{S^c}) \end{bmatrix} \\ &+ \lambda \begin{bmatrix} \text{bvec}(\boldsymbol{Z}(\hat{\boldsymbol\Delta}_S)) \\ \text{bvec}(\boldsymbol{Z}(\hat{\boldsymbol\Delta}_{S^c})) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}\,. \end{aligned} \tag{19}$$

The general approach of [30] (followed in [10, 16, 24, 25]) is to first solve the hypothetical constrained optimization problem with known edgeset S

$$\tilde{\boldsymbol\Delta} = \arg \min_{\boldsymbol\Delta : \boldsymbol\Delta_{S^c} = \mathbf{0}} L_\lambda(\boldsymbol\Delta, \hat{\boldsymbol\Sigma}_x, \hat{\boldsymbol\Sigma}_y) \tag{20}$$

where S^c is the complement of S. Since, by construction, $\tilde{\boldsymbol\Delta}_{S^c} = \mathbf{0}$, in this case (19) reduces to

$$\hat{\boldsymbol\Gamma}_{S,S}\text{bvec}(\tilde{\boldsymbol\Delta}_S) - \text{bvec}((\hat{\boldsymbol\Sigma}_x - \hat{\boldsymbol\Sigma}_y)_S) + \lambda\,\text{bvec}(\boldsymbol{Z}(\tilde{\boldsymbol\Delta}_S)) = \mathbf{0}\,. \tag{21}$$

In the approach of [30], one investigates condition under which the solution $\hat{\boldsymbol\Delta}$ to (4) is the same as the solution $\tilde{\boldsymbol\Delta}$ to (20). This is done by showing that $\tilde{\boldsymbol\Delta}$ satisfies (19). The choice $\hat{\boldsymbol\Delta} = \tilde{\boldsymbol\Delta}$ implies that $\hat{\boldsymbol\Delta}_{S^c} = \mathbf{0}$ and (21) is true with $\tilde{\boldsymbol\Delta}$ replaced with $\hat{\boldsymbol\Delta}$. In order to satisfy (19), it remains to show that for any edge $e \in S^c$,

$$\|\hat{\boldsymbol\Gamma}_{e,S}\text{bvec}(\tilde{\boldsymbol\Delta}_S) - \text{bvec}((\hat{\boldsymbol\Sigma}_x - \hat{\boldsymbol\Sigma}_y)_e)\|_2 < \lambda \tag{22}$$

where for $\boldsymbol{a} \in \mathbb{R}^q$, $\|\boldsymbol{a}\|_2 = \sqrt{\boldsymbol{a}^\top\boldsymbol{a}}$. This requires a set of sufficient conditions, along with additional conditions for performance characterization, which we discuss next.

As in [16], it is convenient to define an operator $\boldsymbol{\mathcal{C}}(\cdot)$ that "operates on block matrices and outputs a smaller matrix with elements that equal to the Frobenius norm of the original blocks," with $\boldsymbol{\mathcal{C}}(\boldsymbol\Sigma^*) \in \mathbb{R}^{p\times p}$ when $\boldsymbol\Sigma^* \in \mathbb{R}^{mp\times mp}$, $\boldsymbol\Sigma^* \in \{\boldsymbol\Sigma_x^*, \boldsymbol\Sigma_y^*\}$, and $\boldsymbol{\mathcal{C}}(\boldsymbol\Sigma_y \boxtimes \boldsymbol\Sigma_x) \in \mathbb{R}^{p^2\times p^2}$ while $(\boldsymbol\Sigma_y \boxtimes \boldsymbol\Sigma_x) \in \mathbb{R}^{(mp)^2\times(mp)^2}$. In particular, $\boldsymbol{\mathcal{C}}(\boldsymbol\Delta^{(k\ell)}) = \|\boldsymbol\Delta^{(k\ell)}\|_F$ and $\boldsymbol{\mathcal{C}}(\boldsymbol\Sigma_y^{(ij)} \otimes \boldsymbol\Sigma_x^{(k\ell)}) = \|\boldsymbol\Sigma_y^{(ij)} \otimes \boldsymbol\Sigma_x^{(k\ell)}\|_F$.

In rest of this section we allow p, s and λ to be a functions of sample size n, denoted as p_n, s_n and λ_n, respectively. Lemma 1 follows from [16, p. 1739] which is based on [30, Lemma 1].

Lemma 1: Suppose $\hat{\boldsymbol\Sigma} = (1/n)\sum_{t=1}^n \boldsymbol{x}(t)\boldsymbol{x}^\top(t)$, given n independent samples $\{\boldsymbol{x}(t)\}_{t=1}^n$ of $\boldsymbol{x} \sim \mathcal{N}(\mathbf{0}, \boldsymbol\Sigma^*)$, $\boldsymbol{x} \in \mathbb{R}^{mp}$. Define $\tilde C_0 = 40m\big(\max_{1\le i\le mp_n}\Sigma^*_{ii}\big)\sqrt{2\big(\tau + \ln(4m^2)/\ln(p_n)\big)}$. Then

$$P\Big(\|\boldsymbol{\mathcal{C}}(\hat{\boldsymbol\Sigma} - \boldsymbol\Sigma^*)\|_\infty > \tilde C_0\sqrt{\ln(p_n)/n}\Big) \le 1/p_n^{\tau-2} \tag{23}$$

for any $\tau > 2$ and $n > 2(\ln(4) + \tau\ln(mp_n))$. •

Using the union bound and Lemma 1, we have Lemma 2.

Lemma 2: Let $\hat{\boldsymbol\Sigma}_x$ and $\hat{\boldsymbol\Sigma}_y$ be as in (2). Define $n = \min(n_x, n_y)$, $\bar\sigma_{xy} = \max\{\max_i \Sigma^*_{x,ii},\ \max_i \Sigma^*_{y,ii}\}$ and

$$\begin{aligned} \mathcal{A} =& \max\left\{\|\boldsymbol{\mathcal{C}}(\hat{\boldsymbol\Sigma}_x - \boldsymbol\Sigma_x^*)\|_\infty\,, \|\boldsymbol{\mathcal{C}}(\hat{\boldsymbol\Sigma}_y - \boldsymbol\Sigma_y^*)\|_\infty\right\} \\ C_0 =& 40\,m\,\bar\sigma_{xy}\sqrt{2\big(\tau + \ln(4m^2)/\ln(p_n)\big)}\,. \end{aligned} \tag{24}$$

Then for any $\tau > 2$ and $n > 2(\ln(4) + \tau\ln(mp_n))$,

$$P\Big(\mathcal{A} > C_0\sqrt{\ln(p_n)/n}\Big) \le 2/p_n^{\tau-2} \quad \bullet \tag{25}$$

Define

$$M = \max\{\|\boldsymbol{\mathcal{C}}(\boldsymbol\Sigma_x^*)\|_\infty\,, \|\boldsymbol{\mathcal{C}}(\boldsymbol\Sigma_y^*)\|_\infty\}\,, \tag{26}$$

$$M_\Sigma = \max\{\|\boldsymbol{\mathcal{C}}(\boldsymbol\Sigma_x^*)\|_{1,\infty}\,, \|\boldsymbol{\mathcal{C}}(\boldsymbol\Sigma_y^*)\|_{1,\infty}\}\,, \tag{27}$$

$$\kappa_\Gamma = \|(\Gamma^*_{S,S})^{-1}\|_{1,\infty}\,, \tag{28}$$

$$\alpha = 1 - \max_{e\in S^c}\|\boldsymbol{\mathcal{C}}(\boldsymbol\Gamma^*_{e,S}(\boldsymbol\Gamma^*_{S,S})^{-1})\|_1 \tag{29}$$

where S and $\boldsymbol{\Gamma}^*$ have been defined in (14) and (18). In (29), we require $0 < \alpha < 1$, and the expression

$$\max_{e \in S^c} \|\boldsymbol{\mathcal{C}}(\boldsymbol{\Gamma}^*_{e,S}(\boldsymbol{\Gamma}^*_{S,S})^{-1})\|_1 \le 1 - \alpha$$

for some $\alpha \in (0,1)$ is called the *irrepresentability condition*. Similar conditions are also used in [10, 16, 24, 25, 30].

Let $\hat{\boldsymbol{\Delta}}$ be as in (5).

Theorem 1 : For the system model of Sec. 2, under the irrepresentability condition (29) for some $\alpha \in (0,1)$, if

$$\lambda_n = \max\left\{\frac{8}{\alpha}, \frac{3}{\alpha \bar{C}_\alpha} s_n \kappa_\Gamma M C_{M\kappa}\right\} C_0 \sqrt{\frac{\ln(p_n)}{n}} \tag{30}$$

$$n = \min(n_x, n_y) > C_0^2 \ln(p_n) \max\left\{\frac{1}{\min\{M^2, 1\}}, 81 M^2 s_n^2 \kappa_\Gamma^2, \frac{9 s_n^2}{(\alpha \bar{C}_\alpha)^2} (\kappa_\Gamma M C_{M\kappa})^2\right\} \tag{31}$$

where $C_{M\kappa} = (3/2)(1 + \kappa_\Gamma M_\Sigma^2)$, then with probability $> 1 - 2/p_n^{\tau-2}$, for any $\tau > 2$, we have

(i) $\|\boldsymbol{\mathcal{C}}(\hat{\boldsymbol{\Delta}} - \boldsymbol{\Delta}^*)\|_\infty \le (C_{b1} + C_{b2}) C_0 \sqrt{\frac{\ln(p_n)}{n}}$

$$\text{where } \; C_{b1} = 3\kappa_\Gamma \max\left\{\frac{8}{\alpha}, \frac{3}{\alpha \bar{C}_\alpha} s_n \kappa_\Gamma M C_{M\kappa}\right\}$$

$$C_{b2} = 9 s_n \kappa_\Gamma^2 M^2, \quad \bar{C}_\alpha = \frac{1-\alpha}{2(2M+1) - 2\alpha M}.$$

(ii) $\hat{\boldsymbol{\Delta}}_{S^c} = \mathbf{0}$.

(iii) $\|\boldsymbol{\mathcal{C}}(\hat{\boldsymbol{\Delta}} - \boldsymbol{\Delta}^*)\|_F \le \sqrt{s_n}\, \|\boldsymbol{\mathcal{C}}(\hat{\boldsymbol{\Delta}} - \boldsymbol{\Delta}^*)\|_\infty$.

(iv) Additionally, if $\min_{(k,\ell) \in S} \|(\boldsymbol{\Delta}^*)^{(k\ell)}\|_F \ge 2(C_{b1}+C_{b2}) C_0 \sqrt{\frac{\ln(p_n)}{n}}$, then $P(\mathcal{G}_{\hat{\Delta}} = \mathcal{G}_{\Delta^*}) > 1 - 2/p_n^{\tau-2}$ (support recovery). •

The proof of Theorem 1 is omitted for lack of space. The main effort is in proving part (i). Parts (iii) and (iv) follow immediately from part (i) (as in [30, Theorem 1]), and part (ii) is a consequence of the fact that $\hat{\boldsymbol{\Delta}} = \tilde{\boldsymbol{\Delta}}$.

Remark 1: Convergence Rate. If M, M_Σ and κ_Γ stay bounded with increasing sample size n, we have $\|\boldsymbol{\mathcal{C}}(\hat{\boldsymbol{\Delta}} - \boldsymbol{\Delta}^*)\|_F = \mathcal{O}_P(s_n^{1.5} \sqrt{\ln(p_n)/n})$. Therefore, for $\|\boldsymbol{\mathcal{C}}(\hat{\boldsymbol{\Delta}} - \boldsymbol{\Delta}^*)\|_F \to 0$ as $n \to \infty$, we must have $s_n^{1.5}\sqrt{\ln(p_n)/n} \to 0$. The single-attribute results in [10] need $s_n^{2.5}\sqrt{\ln(p_n)/n} \to 0$. Recall that $s_n = |S| = |\mathcal{E}_\Delta|$, number of edges in the differential graph. □

5. NUMERICAL EXAMPLE

We consider an Erdös-Rènyi graph where p nodes are connected to each other with probability $p_{er} = 0.5$ and there are $m = 3$ attributes per node. In the upper triangular $\boldsymbol{\Omega}_x$, we set $[\boldsymbol{\Omega}_x^{(jk)}]_{st} = 0.5^{|s-t|}$ for $j = k = 1, \cdots, p$, $s, t = 1, \cdots, m$. For $j \ne k$, if the two nodes are not connected, we have $\boldsymbol{\Omega}^{(jk)} = \mathbf{0}$, and if nodes j and k are connected, then $[\boldsymbol{\Omega}^{(jk)}]_{st}$ is uniformly distributed over $[-0.4, -0.1] \cup [0.1, 0.4]$. Then add lower triangular elements to make $\boldsymbol{\Omega}_x$ a symmetric matrix. To generate $\boldsymbol{\Omega}_y$, we follow [10] and first generate a differential graph with $\boldsymbol{\Delta} \in \mathbb{R}^{(mp)\times(mp)}$ as an Erdös-Rènyi graph with connection probability $p_{er} = 0.05$ (sparse): if nodes j and k are connected, then each of m^2 elements of $\boldsymbol{\Delta}^{(jk)}$ is independently set to ±0.9 with equal probabilities. Then $\boldsymbol{\Omega}_y = \boldsymbol{\Omega}_x + \boldsymbol{\Delta}$. Finally add $\gamma \boldsymbol{I}$ to $\boldsymbol{\Omega}_y$ and to $\boldsymbol{\Omega}_x$ and pick γ so that $\boldsymbol{\Omega}_y$ and $\boldsymbol{\Omega}_x$ are both positive definite. With $\boldsymbol{\Phi}_x \boldsymbol{\Phi}_x^\top = \boldsymbol{\Omega}_x^{-1}$, we generate $\boldsymbol{x} = \boldsymbol{\Phi}\boldsymbol{w}$ with $\boldsymbol{w} \in \mathbb{R}^{mp}$ as Gaussian $\boldsymbol{w} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$, and similarly for $\boldsymbol{y}$. We generate $n = n_x = n_y$ i.i.d. observations for $\boldsymbol{x}$ and $\boldsymbol{y}$, with $m = 3$, $p = 100$, $n \in \{100, 200, 300, 400, 800, 1200, 1600\}$.

Simulation results based on 50 runs are shown in Figs. 1 and 2. By changing the penalty parameter λ and determining the resulting edges, we calculated the true positive rate (TPR) and false positive rate 1-TNR (where TNR is the true negative rate) over 50 runs. The receiver operating characteristic (ROC) is shown in Fig. 1 for our multi-attribute approach (labeled "MA") as well as for a single-attribute approach (labeled "SA"), based on [24], where we first estimate an mp-node differential graph, and then use $\|\hat{\boldsymbol{\Delta}}^{(k\ell)}\|_F \ne 0 \Leftrightarrow \{\{k, \ell\} \in \mathcal{E}_\Delta$. It is seen from Fig. 1 that our approach outperforms the SA approach (that uses the same cost but element-wise lasso penalty instead of group-lasso penalty). In Fig. 2 we show the results based on 50 runs for our approach when BIC parameter selection method (Sec. 3) is applied. Here we show the TPR, 1-TNR and F_1score values along with the $\pm\sigma$ error bars. The proposed approach works well both in terms of F_1 score and TPR vs 1-TNR.

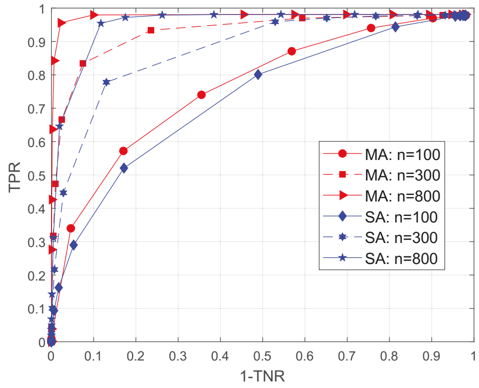

Fig. 1: ROC curves. TPR=true positive rate, TNR=true negative rate

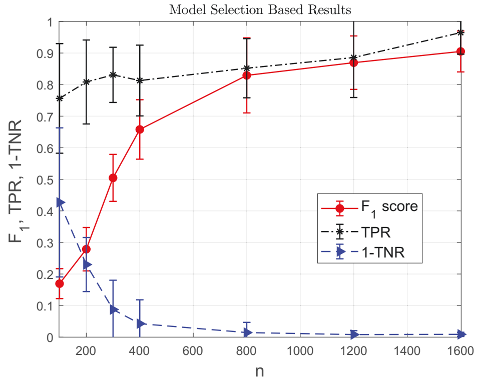

Fig. 2: BIC based results: F_1-scores, TPR and 1-TNR

6. CONCLUSIONS

A group lasso penalized D-trace loss function approach for differential graph learning from multi-attribute data was presented. An ADMM algorithm was presented to optimize the convex objective function. Theoretical analysis establishing consistency of the estimator in high-dimensional settings was performed. We illustrated our approach using numerical examples where the multi-attribute approach is shown to outperform a single-attribute approach in correctly detecting the differential graph edges with ROC as the performance metric.

7. REFERENCES

[1] S.L. Lauritzen, *Graphical models*. Oxford, UK: Oxford Univ. Press, 1996.

[2] J. Whittaker, *Graphical Models in Applied Multivariate Statistics*. New York: Wiley, 1990.

[3] P. Danaher, P. Wang and D.M. Witten, "The joint graphical lasso for inverse covariance estimation across multiple classes," *J. Royal Statistical Society, Series B (Methodological)*, vol. 76, pp. 373-397, 2014.

[4] J. Friedman, T. Hastie and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432-441, July 2008.

[5] N. Meinshausen and P. Bühlmann, "High-dimensional graphs and variable selection with the Lasso," *Ann. Statist.*, vol. 34, no. 3, pp. 1436-1462, 2006.

[6] K. Mohan, P. London, M. Fazel, D. Witten and S.I. Lee, "Node-based learning of multiple Gaussian graphical models," *J. Machine Learning Research*, vol. 15, pp. 445-488, 2014.

[7] B. Zhao, Y.S. Wang and M. Kolar, "Direct estimation of differential functional graphical models," in *Proc. 33rd Conf. Neural Information Processing Systems (NeurIPS 2019)*, Vancouver, Canada, 2019.

[8] B. Zhao, Y.S. Wang and M. Kolar, "FuDGE: A method to estimate a functional differential graph in a high-dimensional setting," *J. Machine Learning Research*, vol. 23, pp. 1-82, 2022.

[9] Y. Wu, T. Li, X. Liu and L.| Chen, "Differential network inference via the fused D-trace loss with cross variables," *Electronic J. Statistics*, vol. 14, pp. 1269-1301, 2020.

[10] H. Yuan, R. Xi, C. Chen and M. Deng, "Differential network analysis via lasso penalized D-trace loss," *Biometrika*, vol. 104, pp. 755-770, 2017.

[11] Z. Tang, Z. Yu and C. Wang, "A fast iteraive algorithm for high-dimensional differential network," *Computational Statistics*, vol. 35, pp. 95-109, 2020.

[12] S.D. Zhao, T.T. Cai and H. Li, "Direct estimation of differential networks," *Biometrika*, vol. 101, pp. 253-268, June 2014.

[13] E. Belilovsky, G. Varoquaux and M.B. Blaschko, "Hypothesis testing for differences in Gaussian graphical models: Applications to brain connectivity," *Advances in Neural Information Processing Systems (NIPS 2016)*, vol. 29, Dec. 2016.

[14] J. Chiquet, G. Rigaill and M. Sundquist, "A multiattribute Gaussian graphical model for inferring multiscale regulatory networks: an application in breast cancer." In: Sanguinetti G., Huynh-Thu V. (eds), *Gene Regulatory Networks. Methods in Molecular Biology*, vol 1883. Humana Press, New York, NY, 2019.

[15] M. Kolar, H. Liu and E.P. Xing, "Markov network estimation from multi-attribute data," in *Proc. 30th Intern. Conf. Machine Learning (ICML)*, Atlanta, GA, 2013.

[16] M. Kolar, H. Liu and E.P. Xing, "Graph estimation from multi-attribute data," *J. Machine Learning Research*, vol. 15, pp. 1713-1750, 2014.

[17] J.K. Tugnait, "Sparse-group lasso for graph learning from multi-attribute data," *IEEE Trans. Signal Process.*, vol. 69, pp. 1771-1786, 2021. (Corrections, vol. 69, p. 4758, 2021.)

[18] G. Marjanovic and V. Solo, "Vector l_0 sparse conditional independence graphs," in *Proc. IEEE ICASSP 2018*, pp. 2731-2735, 2018.

[19] Z. Yue, P. Sundaram and V. Solo, "Fast block-sparse estimation for vector networks," in *Proc. IEEE ICASSP 2020*, pp. 5505-5509, 2020.

[20] P. Sundaram, M. Luessi, M. Bianciardi, S. Stufflebeam, M. Hämäläinen and V. Solo, "Individual resting-state brain networks enabled by massive multivariate conditional mutual information," *IEEE Trans. Med. Imaging*, vol. 39, pp. 1957-1966, 2020.

[21] Z. Yue and V. Solo, "Comparing vector networks via frequency domain persistent homology," in *Proc. IEEE CDC*, pp. 126-131, Dec. 2021.

[22] D.S. Tracy and K.G. Jinadasa, "Partitioned Kronecker products of matrices and applications," *Canadian J. Statistics*, vol. 17, pp. 107-120, March 1989.

[23] S. Liu, "Matrix results on Khatri-Rao and Tracy-Singh products," *Linear Algebra & Its Applications*, vol. 289, pp. 267-277, 1999.

[24] B. Jiang, X. Wang and C. Leng, "A direct approach for sparse quadratic discriminant analysis," *J. Machine Learning Research*, vol. 19, pp. 1-37, 2018.

[25] T. Zhang and H. Zou, "Sparse precision matrix estimation via lasso penalized D-trace loss," *Biometrika*, vol. 101, pp. 103-120, 2014.

[26] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. Royal Statistical Society, Series B (Methodological)*, vol. 68, pp. 49-67, 2006.

[27] J. Friedman, T. Hastie and R. Tibshirani, "A note on the group lasso and a sparse group lasso," *arXiv:1001.0736v1 [math.ST]*, 5 Jan 2010.

[28] N. Simon, J. Friedman, T. Hastie and R. Tibshirani, "A sparse-group lasso," *J. Computational Graphical Statistics*, vol. 22, pp. 231-245, 2013.

[29] S. Boyd, N. Parikh, E. Chu, B. Peleato and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1-122, 2010.

[30] P. Ravikumar, M.J. Wainwright, G. Raskutti and B. Yu, "High-dimensional covariance estimation by minimizing ℓ_1-penalized log-determinant divergence," *Electronic J. Statistics*, vol. 5, pp. 935-980, 2011.